vCenter Configuration Manager Transport Layer Security Implementation

VMware VCM 5.3

WHITE PAPER

Table of Contents

Introduction to TLS

Transport Layer Security (TLS) and its predecessor, Secure Sockets Layer (SSL), are cryptographic protocols that provide endpoint authentication and secure communications over any transport. TLS is normally associated with Internet communication but can be applied to any transport layer, including sockets and HTTP. TLS allows for two levels of security: Server Authentication and Mutual Authentication.

Server Authentication

Server Authentication authenticates the server to the client. When server authentication is used, the end user, or client, verifies that the server they are communicating with is actually who it says that it is. In the Internet world, your browser is the client, and a website such as Amazon™ is the server. Millions of clients need to be able to prove that the site to which they are giving financial information is really Amazon™.

To accomplish this using TLS, Amazon™ provides a certificate issued by a trusted authority, such as Verisign®. If your browser has the Verisign® Certification Authority certificate in its trusted store, it can trust that the server really is Amazon™. Typically, the server authenticates the client/user by asking for authentication information, such as a user name and password.

VCM supports Server Authentication. That is, in VCM environments where TLS is employed, VCM Agents are able to verify the identity of the VCM Collector (or Collectors) through the use and verification of certificates. A description of this process is provided later in this paper.

Mutual Authentication

Mutual Authentication authenticates the server to the client, and the client to the server. When Mutual Authentication is used, both the client and the server provide and validate certificates in order to verify each other’s identity.

VCM is Mutual Authentication ready. This means that Agent certificates can be manually created and registered to create a Mutual Authentication environment. However, VCM does not support this mode out-of-the-box, or supply any functionality to aid in the administration of Agent Certificates. Contact VMware Customer Support for instructions.

Certificates and Public Key Infrastructure

A Public Key Infrastructure, or PKI, is a management system that aids in the administration and distribution of public keys and certificates. TLS can use certificates managed by a public key infrastructure to guarantee the identity of servers and clients. Certificates can be created, managed and used by TLS without a PKI. For more information about manually creating certificates, see Creating Certificates for TLS Using Makecert on page 20.

There are two main types of encryption algorithms:

- Single Key, symmetric, or secret key encryption algorithms use a single key, which must be kept secret.
- Public Key, or asymmetric algorithms use a pair of keys. One key is used to encrypt information, the other to decrypt. The process is reversible. Either key can be used to encrypt. The other must be used to decrypt. Asymmetric encryption is much slower than symmetric encryption. It is common to use an asymmetric protocol to securely negotiate a session key, which is a secret key used only for the duration of a single connection. The public key in a key pair may be freely passed around. However, it is important to verify that you have the key you think you have, and that it belongs to the entity you think it belongs to. Certificates are a mechanism for making this identification.

A certificate is a package containing a public key, information identifying the owner or source of the key, and one or more certifications (or signatures) verifying that the package is authentic.

To sign a certificate, an issuer adds information about itself to the information already in a certificate request. The public key and identifying information are hashed and signed using the private key of the issuer's certificate.

If you have the public key of the issuer, you can verify that the public key in the certificate belongs to the entity identified in the certificate (if you trust the issuer). You will have a certificate for the issuer with the same type of information. The issuer’s certificate is, in turn, signed by another issuer. This is called a certification path, or trust chain. The path ends when you arrive at a certificate that is issued/signed by itself, or when one of the certificates is explicitly trusted. The path is trusted if it ends in a trusted certificate. Typically, this means that someone has installed the certificate in a trusted certificate store.

Expiration and Revocation

Keys and certificates are not designed to be used permanently. Keys can be compromised and circumstances can change. Certificates are created with a certain period of validity, before and after which they should not be used or trusted. If any certificate expires (the "valid-to"/"not after" date passes without renewing or replacing the certificate), then it cannot be used to establish a TLS session.

In addition, certificates can be revoked before they expire to indicate the withdrawal of trust. The issuing authority may make a certificate revocation list (CRL) available as additional validation for certificates it has issued. Any certificates in the list should not be trusted.

To view your VCM certificates at any time in the VCM Portal, click **Administration | Certificates**. The data grid displays your certificates and related information and expiration dates.

For information on how to renew or replace your certificates, see Changing Certificates on page 13.

Note VCM supports certificate expiration. However, it does not support revocation lists. Certificates can be removed from the certificate stores to effectively "revoke" them.

Certificate Standards

Certificates are defined by the X.509 RFC standard. This standard includes certain standard fields and capabilities. Those who implement certificates may add additional fields, which can be marked as either critical or non-critical. These fields are a contract between the creator and consumer. Because they are implementation-defined, an application might encounter a certificate with fields that it does not understand. The application is obligated to fail validation on a certificate with critical extensions that it does not understand. Non-critical extensions may be ignored.

One of the non-critical extensions is Enhanced Key Usage. This extension is used to specify the uses for which the certificate is valid. These usages may include Server Authentication, Client Authentication, Code Signing, and Certificate Signing.

Certificate Storage

In Microsoft® systems, certificates are stored in certificate stores. Certificate stores may be located in files, the registry, memory, Active Directory, and other locations. Logical certificate stores provide a unified view of a collection of physical stores that share common properties. All discussion of Microsoft certificate stores in this document refers to logical stores. For a description of the logical system stores provided by Microsoft, see Microsoft TechNet: Certificate Stores.

On UNIX systems, Collector Certificates (for Server Authentication) and Agent Certificates and Agent private keys (for Mutual Authentication) are stored in a proprietary protected store. Although this store is not encrypted, it is protected from simple viewing. Use the CSI_ManageCertificateStore utility and the associated help provided with your VCM UNIX Agent installation package to view or manage the UNIX Agent Certificate store. For more information, see the *VMware vCenter Configuration Manager Installation and Getting Started Guide*.

All VCM Agents using HTTP should be able to trust any VCM Collector Certificate, not just the Collector that the Agent installation package was generated on. This may be via an Enterprise Certificate or through an existing PKI system. If this environmental requirement is not met, only the Collector that generated the Agent installation package (and any Collectors that share an Enterprise certificate with that Collector) will be able to communicate with the Agent using the HTTP protocol. See How VCM Uses Certificates on page 7.

All VCM Collectors should be able to trust any Agent Certificate, even those issued by other Collectors. This may be via an Enterprise Certificate or an existing PKI system. It this requirement is not met, the Collector will fail to establish Mutual Authentication with Agents that are certified by another Collector.

How VCM Uses Certificates

There are three types of certificates that enable HTTP collector-agent communications in VCM:

- Enterprise Certificate
- One or more Collector Certificates
- Agent Certificates for each Agent (used in optional Mutual Authentication)

Certificate information regarding the Enterprise and Collector certificates is collected in VCM. See **Administration | Certificates**.

The Enterprise Certificate

The Enterprise Certificate enables VCM to operate in a multi-collector environment. Agents have the Enterprise Certificate in their trusted certificate stores, and can use it implicitly to validate any certificate issued by the Enterprise Certificate. All Collector Certificates are expected to be issued by the Enterprise Certificate. This is critical in environments where a single Agent is shared between two Collectors.

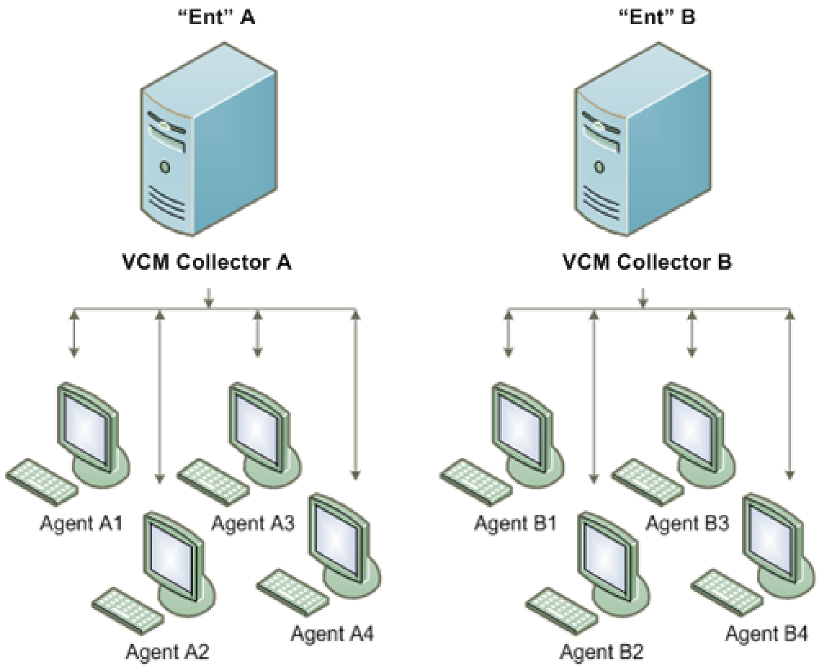

Figure 1: Dedicated Collector-Agent Relationship

The diagram above illustrates a dedicated Collector-Agent relationship. This type of environment includes two Collectors (Collector A and Collector B) that each have a dedicated set of Agents that they collect from. Each Agent has its Collector's Issuer (Enterprise) certificate.

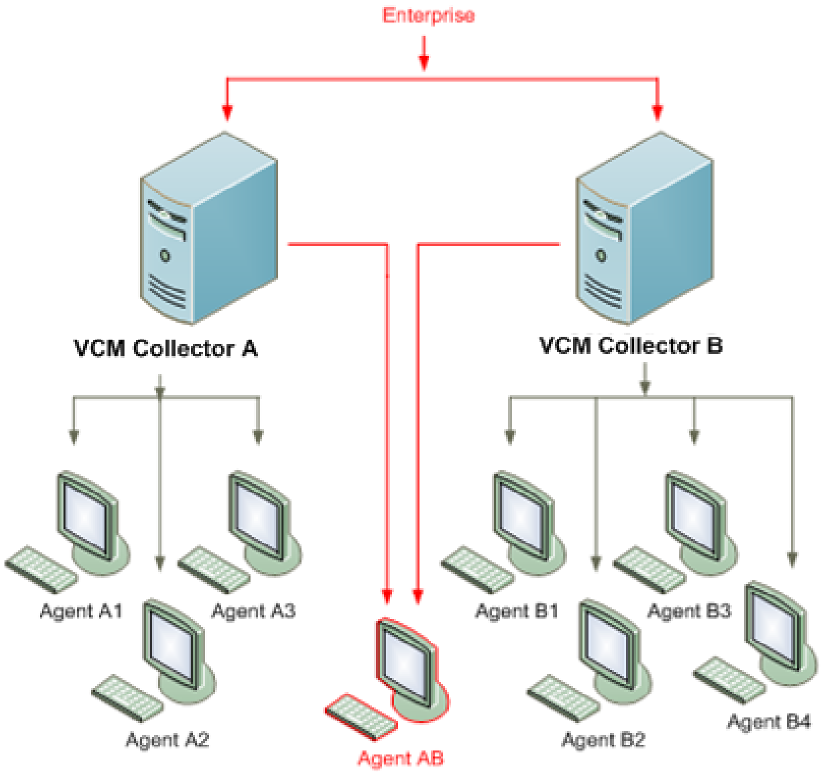

Figure 2: Shared Collector-Agent Relationship

As the diagram above illustrates, an Agent may communicate with more than one Collector. In this case, each Collector has a common Enterprise Certificate. Because both of the Collector certificates were issued by the same trusted authority, the Agent that is shared between the two can trust both Collector Certificates. This is useful in multi-collector, Server Authentication environments.

If you employ Mutual Authenticate and if a single Agent is shared between two Collectors, then it becomes necessary for each Collector to use an Agent Certificate that is issued by a Collector other than itself. This is described as a shared Collector-Agent relationship.

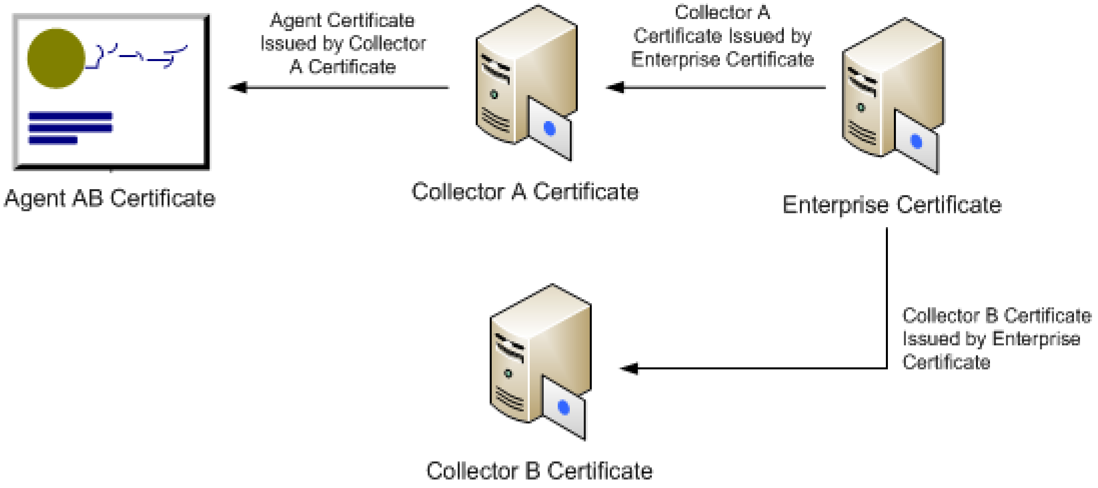

Figure 3: Trust Chain in a Shared Collector-Agent Relationship

In addition, for Mutual Authentication in a shared Collector-Agent relationship, each Collector trusts the Agent Certificate because that Agent Certificate was issued by a Collector Certificate which was, in turn, issued by the trusted Enterprise Certificate. Since both Collectors trust the Enterprise Certificate, then they can also trust the Agent Certificate that was issued by a Collector Certificate that was issued by the Enterprise Certificate.

Enterprise Certificates in VCM must have the following properties:

- Must be capable of signing certificate requests.
- The path length basic constraint, if present, must be at least two if the Collector certificate will be used for issuing Agent certificates. This means that the Enterprise Certificate may issue a Collector Certificate that may issue Agent Certificates.
- May be self-signed. If the certificate is self-signed, it will have to be trusted itself. Trust is bestowed by placing the certificate in the Trusted Root store (Windows) or in the VCM store (UNIX). This represents a VCM-specific trust chain.
- May be signed by another certificate in an existing PKI and placed in the trusted store.
- Must be stored in the local machine Trusted Root Certification Authorities store on the Windows Collector and Agents (Windows only).
- On UNIX platforms, the Agent has a vendor-implemented certificate store. The Enterprise Certificate(s) must be added to this store. One will be included during initial installation, and subsequent certificate(s) must be added manually using the CSI_ManageCertificateStore utility included with your VCM UNIX Agent installation.

The Collector Certificate

The Collector Certificate is issued by the Enterprise Certificate, and must be usable for Server Authentication and, optionally, certificate signing (also known as “issuing”). Server Authentication is required to establish a TLS connection with an Agent. Certificate signing is required to issue Agent Certificates for Mutual Authentication. It is technically possible to split these functions between two certificates or two Collectors.

The Collector Certificate is used to initiate and secure a TLS communication channel with an HTTP Agent. The Agent must be able to establish that the Collector Certificate can be trusted. That means that the Collector Certificate is valid and the certification path starting with the Collector Certificate must end in a trusted certificate. By design, the Enterprise Certificate will be installed in the Agent's trusted store, and the chain will end with the Enterprise Certificate.

The Collector Certificate may also be used to issue Agent Certificates. As long as all Collector Certificates are issued by the same Enterprise Certificate, any Agent Certificate may be issued by any Collector Certificate, and all Collectors will be able to trust all Agents. Similarly, all Agents will be able to validate all Collector Certificates.

The Collector Certificate and associated private key must be available to the Collector. This certificate will be stored in the (local machine) personal system store. Collector Certificates in VCM must adhere to the following requirements:

- Must be located in the local machine personal certificate store of the Collector.
- Must be valid for Server Authentication (OID: 1.3.6.1.5.5.7.3.1).
- If the Collector certificate will be used to issue Agent certificates for mutual authentication:
    - If the key usage extension is present, it must include certificate signing.
    - Must be an authority rather than an end certificate.
    - If the path length is set on the basic constraints, it must be greater than or equal to 1.

Agent Certificates

Agent Certificates are used only in Mutual Authentication. The Agent machine either produces a certificate request, or one is manually produced on the Agent's behalf. A Collector issues a certificate based on this request. Copies of the certificate are stored on both the Agent machine and the Collector. The Agent's private key should never exist anywhere but on the Agent machine.

When a second Collector contacts the agent, the Agent makes its certificate available, and the second Collector also stores the certificate. Note that a second Collector cannot renew an Agent Certificate that was issued by a previous Collector.

Certificates are also used to encrypt and distribute the ESX and ESXi Web Service credentials (Virtualization) and to encrypt and distribute the VCM for Service Desk Integration credentials. These certificates may be the same Agent Certificates used for TLS.

Agent certificates in VCM must adhere to the following requirements:

- Must be usable for client authentication
- Must be issued by any Collector Certificate issued by the Enterprise Certificate, known to the Agent

TLS Machine Security Level

Once the Collector establishes communication with an Agent using TLS, the Collector does not permit HTTP communication without it. To do so would allow a malicious actor to impersonate either the Collector or Agent by downgrading the communication security level.

The restrictions concerning the establishment of Server Authentication and Mutual Authentication relationships are:

- Once an agent has established Server Authentication, the Collector will not allow non-TLS HTTP communication.
- Once an agent has established Mutual Authentication, the Collector will not allow non-TLS HTTP or Server (only) Authenticated TLS communication.
- The Collector supports both TLS and non-TLS capable Agents from earlier releases. Please contact VMware Customer Support for assistance using the current release with earlier Agents (TLS and non-TLS enabled).

These restrictions do not apply to DCOM. The Security level persists across change protocol and installation/upgrade actions.

Creating and Installing Certificates for Collectors

Certificates can either be generated during VCM installation, or created in advance of installation and stored in the local certificate store.

When you select either of these options, the VCM Installation Manager will automatically register the selected certificates in VCM and configure the Agents to trust these certificates.

Installation of Certificates to Collectors

VCM Installation Manager offers you the options of either generating your certificates during installation (see *VMware vCenter Configuration Manager Installation and Getting Started Guide*) or browsing to your certificate store to select pre-generated certificates.

If you will be providing your own pre-generated certificates, they must exist on the Collector machine prior to VCM Installation. The Collector Certificate must be in the Local Machine Personal system store, and the Enterprise Certificate must be in the Local Machine Trusted Root system store. The private key of the Enterprise certificate does not need to be available. The certificates do not need to be available on the database machine in a split configuration. The install interview will prompt the you for the names of the certificates to be used.

- Generating Certificates During Installation: During VCM installation, the VCM Installation Manager allows you to generate your Collector and Enterprise Certificates during the installation process. For more information about generating certificates during VCM installation, refer to the *VMware vCenter Configuration Manager Installation and Getting Started Guide*.
- Creating Certificates Prior to Installation: If you want to create your own certificates in advance of VCM installation, refer to The Collector Certificate on page 10 for requirements or to Creating Certificates for TLS Using Makecert on page 20 if you are creating your own certificates without PKI support. Once your certificates are created, you can select them during the installation process.

After VCM installation, if you decide that you want to use different certificates than the ones that you either generated or selected during the installation process, you must replace those certificates. For more information on replacing certificates, see Changing Certificates on page 13.

Installation of Certificates to Additional Collectors

All Collector Certificates in a customer environment should be issued by the same Enterprise Certificate to ensure seamless operation across Agents and Collectors. Generating certificates for more than a single Collector during installation fails to create this relationship. Just as the VCM certificates were expected to be in place prior to the installation of the first Collector, the VCM certificates must also be installed on subsequent Collectors prior to installation. Each Collector needs its own Collector Certificate, and access to the Enterprise Certificate. If all HTTP Agents are to be contacted by only a single Collector, then a single trust hierarchy is not strictly necessary.

If you plan to use more than one Collector for the same Agent machine, you must establish a parent-to-multiple-Collector children relationship, and cannot use generated certificates on additional Collectors. Contact Customer Support for further details and assistance.

Changing Certificates

Certificates always have an expiration date, after which they are no longer valid. The validity period for a certificate is a matter of policy and ranges from minutes to decades. In these cases, you can either renew or replace your certificates.

Renewing Certificates

Renewal of a certificate means extending the validity period for the certificate, using the same key pair, issuer, and identifying information. Whatever mechanism was used to create the VCM certificates can be used to renew them.

It is possible to renew a certificate by updating the expiration date. In this case, a new certificate is issued with the same public key and identifying information as the old certificate. Since the only change is the validity period, it is safe to accept the new certificate at the same level of trust as the old one. Both certificates are valid for the same purposes, and are both are usable in keeping with their validity periods.

When the Collector initiates communication with the Agent, it sends the certification path from the Collector Certificate to its trusted root (typically the Enterprise Certificate) to the Agent. For each certificate in the path, the Agent checks to see if it has a matching certificate in local machine personal or root stores. If it finds a match in either location and the ”new” certificates have different dates, the Agent will install the new certificates. The current trust level is preserved. No certificate will be added to the trusted store unless an equivalent certificate is already present. The old certificates are not removed.

Replacing Certificates

The only way to ensure the authenticity of a new root or trusted certificate is to receive it from a secure and trusted source. During installation, VCM Installation Manager handles Enterprise and Collector Certificate installation and management. You can either select your own certificates from your certificate store, or have VCM generate the certificates automatically for you. In both cases, your VCM Agents will be automatically configured to properly trust the certificates. In addition, Enterprise and Collector Certificates with updated begin and end times will be automatically added to the Agents' certificate stores as described in Renewing Certificates on page 13.

However there are certain circumstances that may require you replace your Enterprise and Collector Certificates, including:

- Compromised private keys
- Company security policy governing the lifetime of keys
- Company or department changes merging VCM environments
- Product evaluations that previously used VCM-generated certificates that are moved into production without re-installation

Use the following procedures to replace both the Enterprise and Collector Certificates, or just the Collector Certificate, and then install them into the certificate stores of the VCM Collector and Agents.

Replace the Enterprise and Collector Certificates

After VCM installation, if you decide that you want to use different certificates than the ones that you either generated or selected during the installation process, you must replace those certificates.

Use the following procedure to replace both the Enterprise and Collector Certificates.

1. Create or obtain a new Enterprise certificate. For information on how to create an Enterprise Certificate using MakeCert Certificate Creation Tool, see Creating Certificates for TLS Using Makecert on page 20.
2. Create or obtain a new Collector Certificate that is signed by the Enterprise Certificate. For information on how to create a Collector Certificate using MakeCert Certificate Creation Tool, see Creating Certificates for TLS Using Makecert on page 20.
3. Import the Enterprise Certificate into the Local Computer Trusted Root store on the VCM Collector. For more information, see Certificate Transport on page 17.
4. Import the Collector Certificate and the private key into the Personal store on the VCM Collector. For more information, see Certificate Transport on page 17.
5. Update the Collector Certificate thumbprint in the VCM Collector database. For more information, see Updating the Collector Certificate Thumbprint in the VCM Collector Database on page 26
6. Restart the Collector service.
7. Import the Enterprise Certificate into the Trusted Root store on the VCM Windows Agent systems (see Certificate Transport on page 17), install the VCM Agent with the "Enable HTTP" option selected, or change protocol to DCOM and back to HTTP (only if the Collector can communicate with Agents using DCOM protocol). On UNIX Agents, place the certificates into the VCM Agent Certificate store.

Replace Only the Collector Certificates

After VCM installation, you may find that you want to use a different Collector Certificate than you specified during installation, but your Enterprise Certificate is still valid. In this situation, you can use the following procedure to replace only the Collector Certificate.

1. Create or obtain a new Collector Certificate (and associated private key) that is signed by the Enterprise Certificate. For information on how to create a Collector Certificate using MakeCert Certificate Creation Tool, see Creating Certificates for TLS Using Makecert on page 20.
2. Import the Collector Certificate and the private key into the Personal store on the VCM Collector.
3. Update the Collector Certificate thumbprint in the VCM Collector database. See Updating the Collector Certificate Thumbprint in the VCM Collector Database on page 26.
4. Restart the Collector services.

Delivering Initial Certificates to Agents

VCM Agents use Enterprise Certificates to validate Collector Certificates. Therefore, the Agent must have access to the Enterprise Certificate as a trusted certificate. In most cases, VCM will deliver and install the Enterprise Certificate as needed.

Installing the Agent from the Collector

The Enterprise Certificate is stored in the CollectorData folder on the Collector. The Enterprise Certificate is installed when the VCM Agent is installed or upgraded with the HTTP protocol.

New Installations

In a new Agent installation, all module files will be installed. The Enterprise Certificate will be installed if and when the EcmComSocketListenerService module is installed. If the “Enable HTTP” option is not chosen for the installation, then the module and certificate will not be installed.

Upgrades

All upgrades of HTTP-enabled Agents from non-TLS Agents to TLS Agents receive a new version of the EcmComSocketListenerService, and the Enterprise Certificate. This also applies to upgrades via the “License and Install Agent on Discovered Machines” Discovery rule option (see VCM Help for more information on VCM Discoveries.”

Changing Protocols from DCOM to HTTP

Changing protocols to HTTP causes the EcmComSocketListenerService module to be installed. Since a secure DCOM connection with the Agent exists, the current Enterprise Certificate can be delivered with the EcmComSocketListenerService module.

Changing Protocol from HTTP to DCOM

The EcmComSocketListenerService module is uninstalled from the Agent during this operation. Since DCOM does not use certificates, the Agent will simply stop using them. Any changes to the Enterprise Certificate will not be automatically propagated to the Agent until HTTP is reinstated. Changing protocol from DCOM to HTTP in the future will deliver the current Enterprise Certificate.

Installing the Agent from a Disk (Windows only)

The VCM installation image/DVD does not contain customer-specific certificates. The manual VCM installer requests the location of the Enterprise Certificate file prior to installing. You must have this file available at installation time. The Certificate file (with a .pem extension) can be copied from the CollectorData folder of the Collector. This will be the case whether you run the manual installer directly (CMAgentInstall.exe or the .msi installation package) or use the “Agent Only” option from the DVD autorun program.

Using CMAgtInstall.exe via Network Share to Install the Agent (Windows only)

CMAgtInstall.exe is the manual Agent installer program. As above, the manual installer will request the location of the Enterprise Certificate file before installing. You must have this file available at installation time. The certificate file can be copied from the CollectorData folder of the Collector. This applies to the .msi installation as well.

UNIX/Linux or Mac OS X

Each UNIX/Linux or Mac OS X installation package is targeted for one or more supported platforms. Because the Enterprise and Collector Certificates are embedded within the Agent installation package (if they were specified when the Collector was installed), they are automatically inserted into the UNIX Agent Certificate store during the Agent installation process.

To manage the VCM UNIX Agent Certificate store, use the CSI_ ManageCertificateStore utility and related help provided with your UNIX Agent installation package. For more information about UNIX/Linux or Mac OS X agent installation or packages and platforms, refer to the *VMware vCenter Configuration Manager Installation and Getting Started Guide*.

Installing the Agent Using a Provisioning System

Windows

The manual installation program is available in .exe and .msi formats. Both versions allow the Enterprise Certificate file to be specified with a command line switch. The certificate installation step may also be omitted with a command line switch. When these programs are run through a provisioning system, you must ensure that the Enterprise Certificate is available (and still secure), and configure the program options appropriately.

Alternatively, you may choose to push the Enterprise Certificate to Agents by some other means and configure the provisioning system to omit certificate installation.

UNIX/Linux or Mac OS X

Each UNIX/Linux or Mac OS X installation package is targeted for one or more supported platforms. To install the UNIX/Linux or Mac OS X Agent using a provisioning system, extract the installation package as appropriate and then deploy the extracted file with the provisioning system. The Enterprise Certificate is embedded in the installation package. For more information about UNIX/Linux or Mac OS X Agent installation, refer to the *VMware vCenter Configuration Manager Installation and Getting Started Guide*.

Certificate Expiration

If any certificate expires (the valid-before date passes without renewing or replacing the certificate), then it cannot be used to establish a TLS session. To verify the expiration dates of your certificates, consult the Certificates grid in VCM (**Administration | Certificates**).

Certificate Transport

A certificate contains the public half of a key pair and identifying information, and an authenticating signature. Though none of this information is ”secret”, the information should still be protected.

A certificate can be stored in a format that includes the private key. When this is the case, the data is secret and must be safeguarded, stored, and transported securely.

Note The following information pertains to Windows platforms only. To import or export a certificate to UNIX, use the CSI_ ManageCertificateStore utility provided with your VCM UNIX Agent installation package.

Exporting Certificates (Windows Only)

One way to export certificates is to use Microsoft® Management Console (MMC), as shown in the following procedure.

Using MMC to Export Certificates

1. Open Microsoft Management Console (click **Start | Run**, and then enter **MMC**).
2. Click **File | Add/Remove Snap-in**.
3. Click **Add**, and then select **Certificates** from the Add Standalone Snap-in dialog box.
4. Click **Add.**
5. Select the computer account for the local machine for which the snap-in will manage certificates, and then click **Finish**.
6. Click **Close**, and then click **OK**.
7. Navigate to the certificate store from which you want to export a certificate.

8. Select the certificate to be exported. Right-click, and then select **All Tasks | Export**.
9. The **Certificate Export** wizard appears. Click **Next**.
10. The **Export Private Key** dialog box appears. If the private key for this certificate is available and is marked as exportable, you will have the option of exporting the private key.
11. The **Export File Format** dialog box appears. If you are exporting the private key, you must use the *.pfx format. If not, choose the Base-64 encoded X.509 format. This creates a file that can be imported by several VCM tools.
12. If you are exporting the private key, you will be prompted to enter a password. This password is intended to protect your private key during transport. The password will be required to import the file. If you leave the password fields blank, no password is required on import.
13. The File to Export dialog box appears. Click **Browse** to navigate to a file. If you are exporting the private key, the location should be secure.
14. The **Summary** dialog box appears. Verify your settings, and then click **Finish**.

Importing Certificates (Windows Only)

There are two ways in which you can import a certificate. Your machine may be set up with file associations that allow you to view and import certificate files. This method imports certificates to the appropriate store for the current user. Because VCM expects to find certificates in the Local Computer stores, the certificates would have to be moved. Although MMC allows you to move certificates with drag-and-drop, it doesn’t work correctly on some versions of the operating system.

The second way to import certificates uses MMC and the import wizard with an explicit location, as shown in the following procedure.

Using MMC to Import Certificates

1. Open Microsoft Management Console (click **Start | Run**, and then enter **MMC**).
2. Click **File | Add/Remove Snap-in**.
3. Click **Add**, and then select **Certificates** from the **Add Standalone Snap-in** dialog box.
4. Click **Add**.
5. Select the computer account for the local machine for which the snap-in will manage certificates, and then click **Finish**.
6. Click **Close**, and then click **OK**.
7. Navigate to the local computer certificate stores, and then right-click the store to which you will be importing certificates. Select **All Tasks | Import**.
8. The **Certificate Import Wizard** appears. Click **Next**.

9. The **File to Import** dialog box appears. Select the file to import. Either format is acceptable: *.pfx or *.cer. The *.pem format is typically a synonym for *.cer and is used more commonly on UNIX systems.

10. If the file contains a private key, you will be prompted for a password. If the file contains a private key, you will have the option of marking it as exportable. If you do not, the private key cannot be exported from the system (you will still have the file though). Do NOT Enable Strong Protection.

11. Verify that the certificate will be stored in the correct location, and then click **Next**.

12. The **Summary** dialog box appears. Verify your settings, and then click **Finish**.

Appendix A: Creating Certificates for TLS Using Makecert

VCM is designed to run in TLS mode with two levels of certificates. In this mode, an Enterprise Certificate is the ultimate trusted authority. All Collector Certificates will be signed by this Enterprise Certificate. All Agents will have access to the Enterprise Certificate as a trusted authority. Any Collector Certificate can be used to sign an Agent Certificate. A given Agent should be able to mutually authenticate with multiple Collectors.

In the following process, the Enterprise machine can be the same as the Collector machine. Some of the steps can be simplified if they are the same, or if the Enterprise or Collector machines are set up to be certificate servers. The example is written for the case when the Enterprise machine is also the (first) Collector. When this is not the case, you will have to follow the steps for creating a second collector for the initial collector.

Makecert (Certificate Creation Tool), cert2spc (Software Publisher Certificate Test Tool), pvkimprt (PVK Digital Certificate Files Importer), and many related utilities are available as part of the SDK download from Microsoft. For more information, visit the Microsoft Developer Network and search for the downloads by platform (pre-Vista or Vista):

- Pre-Vista: Windows Server 2003 SP1 Platform SDK full download
- Vista: Windows SDK for Windows Server 2008 and .NET Framework version 3.5

Create the Enterprise Certificate and the First Collector Certificate

Use the following procedure to create the Enterprise Certificate and the first Collector Certificate. Refer to MakeCert Options on page 23 for a list of the options used below and their definitions.

Example:

```
makecert -pe -n "CN = CM Enterprise Certificate AAAAAAAA-AAAA-AAAA-AAAAAAAAAAAAAAAA"
-ss Root -sr LocalMachine -r -sky exchange -sk "CM Enterprise Certificate AAAAAAAA-
AAAA-AAAAAAAA-AAAAAAAAAAAA" -len 1024 -h 2 -cy authority -eku 1.3.6.1.5.5.7.3.1
```

Note VCM embeds a GUID ("AAAAAAAA-AAAA-AAAAAAAA-AAAAAAAAAAAA" or "BBBBBBBB-BBBB-BBBB-BBBBBBBBBBBBBBBB") into the Common Name by convention to ensure that the name is unique; however, this is not a requirement

1. Use the following command to create the CM Enterprise Certificate:

   ```
   makecert -pe -n "<enterprise_cert_name>" -ss Root -sr LocalMachine -r -sky
   exchange -sk "<enterprise_key_name>" -b mm/dd/yyyy -e mm/dd/yyyy -len 1024 -h
   2 -cy authority -eku 1.3.6.1.5.5.7.3.1 <filename[.cer | .pem]>
   ```

2. Use the following command to create the first Collector Certificate, signed by the Enterprise Certificate.

   ```
   makecert -pe -n "<collector_cert_name>" -ss My -sr LocalMachine -sky exchange
   –sk <collector_cert_name> -b mm/dd/yyyy -e mm/dd/yyyy -len 1024 –in
   <enterprise_cert_common_name> -is Root -ir LocalMachine -cy authority
   <collector_cert_name.[cer|pem]>
   ```

Note If the Enterprise Certificate is not stored (with private key) on the Collector, follow the steps below for additional Collector Certificates in Create Certificates for Additional Collectors on page 21.

Create Certificates for Additional Collectors

If additional Collectors are needed, a slightly different process is required to generate the additional Collector Certificates, issued by the Enterprise Certificate. This process can be followed even if the original certificates were generated by the VCM Installation Manager.

Use the following procedure to create an additional Collector Certificate, signed by the Enterprise Certificate. This procedure must be executed on the Enterprise machine (probably the initial Collector), because access to the private key for the Enterprise certificate is required.

The goal is to create an installable file that includes the new Collector's private key without storing that key in the key store of the initial Collector/Enterprise machine. A better way to do this is to generate a key pair and certificate request on the additional Collector machine, and only transport that.

Refer to MakeCert Options on page 23 for a list of the options used below and their definitions.

1. Enter the following command:

   ```
   makecert -pe -n "<collector_cert_name>" -sky exchange -sv "<collector_cert_
   key_file>" -b mm/dd/yyyy -e mm/dd/yyyy -len 1024 -in "<enterprise_cert_common_
   name>" -is Root -ir LocalMachine -cy authority -eku 1.3.6.1.5.5.7.3.1 "
   <collector_cert_name.[pem|cer]>"
   ```

Example:

```
makecert -pe -n "CN=CM Collector Certificate BBBBBBBB-BBBB-BBBB-BBBB-
BBBBBBBBBBBB" -sky exchange -sv "CM Collector BBBBBBBB-BBBB-BBBB-BBBB-
BBBBBBBBBBBB.pvk" -b 04/07/2008 -e 04/07/2018 -len 1024 -in "CM Enterprise
Certificate AAAAAAAA-AAAA-AAAA-AAAAAAAAAAAAAAAA" -is Root -ir LocalMachine -cy
authority -eku 1.3.6.1.5.5.7.3.1 "CM Collector BBBBBBB-BBBB-BBBB-BBBB-
BBBBBBBBBBBB.pem"
```

2. Enter the following command to convert the x509 certificate file to a file-based certificate store in the named .spc file.

   ```
   cert2spc <collector_cert_name>.cer <collector_cert_name>.spc
   ```

   Example:

   ```
   cert2spc "Collector Certificate BBBBBBBB-BBBB-BBBB-BBBB-BBBBBBBBBBBB.cer"
   "Collector Certificate BBBBBBBB-BBBB-BBBB-BBBB-BBBBBBBBBBBB.spc"
   ```

3. Enter the following command to export the file-based certificate store (containing our certificate) and the private key in the key file to a PFX file.

   ```
   pvkimprt -pfx <collector_cert_name>.spc <collector_cert_key_file>
   ```

   This launches the Certificate Export Wizard. Select Yes, export the private key. Keep the .pfx format. Uncheck all of the checkboxes. Optionally choose a password for secure transport of the file (recommended).

   Example:

   ```
   vkimprt -pfx "CM Collector Certificate BBBBBBBB-BBBB-BBBB-
   BBBBBBBBBBBBBBBB.spc" "CM Collector Certificate BBBBBBBB-BBBB-BBBB-
   BBBBBBBBBBBBBBBB.pvk"
   ```

4. Remove your temporary files, especially the key file.

5. Transport the .pfx file containing the new Collector Certificate, and the Enterprise Certificate export file to the new Collector machine.

   The Enterprise Certificate file is located in the CollectorData folder of the initial collector (typically C:\Program Files\VMware\VCM\CollectorData) or you can export it from the local machine trusted root system store. The export file has a .pem extension.

Import the Certificates on the Collector Machines

Perform the following procedure on the new Collector machine, prior to running VCM Installation Manager.

Important: If you are performing this procedure as part of a certificate replacement process, you must manually install the Enterprise and Collector Certificates in the Collectors' certificate stores, and the Enterprise Certificate in the Agents' certificate stores. See Creating and Installing Certificates for Collectors on page 12 and Delivering Initial Certificates to Agents on page 15.

1. Import the Enterprise Certificate into the local machine trusted root store.
2. Import the Collector Certificate into the local machine personal store.
3. The Enterprise and Collector Certificates are now available for use in the VCM installation.

MakeCert Options

Refer to the following table for a list of the options used in the previously described MakeCert commands, and their definitions.

Note The strings: AAAAAA-AAAAAA... and BBBBBBBB-BBBBB... represent GUIDS. VMware uses GUIDS to help create unique names. GUIDS are a useful convention for programmatically creating uniqueness and are generally not necessary in a manual process.

Field	Definition
-b, -e	Specify begin and end dates. Choose appropriate dates, or omit them. **Note** You cannot enter a time with the date. The time will be 12:00AM GMT. If you chose today's date, it will probably refer to early this morning.
-cy authority	Certificates are either "authority" or "end". End certificates are not allowed to sign other certificates.
-eku 1.3.6.1.5.5.7.3.1	The Server Authentication OID, required only for the Collector Certificate.
<filename>	Optional export file name. This file will not contain the private key. The file should have a .cer or .pem extension.

-h 2	Max height of certificate chains. A value of 2 for the Enterprise allows it to sign a Collector certificate capable of signing Agent certificates.
-in <name>	The name of the signing certificate. This would be the common name (CN field) of the Enterprise Certificate when creating Collector certificates.
-ir LocalMachine	The account of the signing certificate. VCM and the examples in this document use "LocalMachine"
-is Root	The location of the signing certificate. "Root" is the trusted root store.
-len	Key length (optional).
-n <collector_cert_ name>	The subject of the Collector certificate. Must be a valid x509 identifier. Collector Certificates generated by the VCM installer will have the form: “CN=VMware VCM Collector Certificate AAAAAAAA-AAAA-AAAA-AAAA-AAAAAAAAAAAA, T=VMware VCM Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, O=<customer_name>” • CN: Generic name based on a GUID generated for each set of certificates created. This field is required. • T: Static field identifying VCM generated certificates and is the same for all generated certificates. This field is optional. • OU: Static field. This field is optional. • O: Contains the customer name identified in the license file. This field is optional.
-n <enterprise_cert_ name>	The subject of the Enterprise Certificate. Must be a valid x509 identifier. Enterprise Certificates generated by the VCM installer will have the form: “CN=VMware VCM Collector Certificate AAAAAAAA-AAAA-AAAA-AAAA-AAAAAAAAAAAA, T=VMware VCM Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, O=<customer_name>” • CN: Generic name based on a GUID generated for each set of certificates created. This field is required. • T: Static field identifying VCM generated certificates and is the same for all generated certificates. This field is optional. • OU: Static field. This field is optional. • O: Contains the customer name identified in the license file. This field is optional.

-pe	Make the private key exportable.
-r	Self sign the certificate.
-sk <collector_key_name>	Name the key container, for easy reference later. This name does not need to be related to the certificate name.
-sk <enterprise_key_name>	Names the key container, for easy reference later. This name does not need to be related to the certificate name.
-sky exchange	Use the key exchange keypair (rather than the signature keypair).
-sr LocalMachine	Specifies the subject's certificate store location. VCM and the examples in this document use "LocalMachine"
-ss My	Specifies the subject's certificate store name that stores the output certificate. "My" designates the personal certificate store.
-ss Root	Specifies the subject's certificate store name that stores the output certificate. "Root" designates the Trusted Root certificate store.
-sv collector_cert_key_file	Store the private key in a file instead of the CSP. The extension is usually .svk or .pvk.

Appendix B: Updating the Collector Certificate Thumbprint in the VCM Collector Database

1. Within MMC, navigate to the Collector Certificate.
2. Right-click the certificate, and then select **Open**. The Certificate Information window appears.
3. Click the Details tab, and then scroll down to the **Thumbprint** field. Copy the value for use in the SQL script shown below.
4. Use the following SQL script to update the Enterprise Certificate in the VCM Collector database (replace "68 32 d7 fd 4d 9d 29 ba ac 0c 2c 90 8f 64 4b 52 d8 b0 16 0d" with your Collector Certificate's thumbprint.

```
use <insert your VCM SB name here>

update ecm_sysdat_configuration_values

set configuration_value = upper(replace(

'68 32 d7 fd 4d 9d 29 ba ac 0c 2c 90 8f 64 4b 52 d8 b0 16 0d'

,' ',''))

where configuration_name='config_security_certificate_fingerprint'
```

Appendix C: Managing the VCM UNIX Agent Certificate Store

The VCM UNIX Agent certificate store is a protected data storage area that is designed to hold enterprise and collector certificates for server authentication, and to hold the agent certificate and private key for mutual authentication. Although this store is not encrypted, it is protected from simple viewing.

Much of the interaction with the VCM UNIX Agent certificate store is taken care of for the user. VCM UNIX installation packages get updated with the enterprise certificate if one is specified when the collector is installed. This certificate is automatically inserted into the certificate store during the VCM UNIX Agent installation process. Also, the user can specify an alternate certificate directory during the VCM UNIX Agent installation if desired.

Additionally, if VCM Collector certificates are updated with extended begin/end dates, in many cases the new certificate will be automatically added to the store.

Using CSI_ManageCertificateStore

The CSI_ManageCertificateStore command-line tool is provided for manual management of the VCM UNIX Agent certificate store. It helps the user to view and modify the contents of the store.

The following documentation assumes the UNIX VCM agent was installed to the default location (/opt/CMAgent). If this is not the case, please adjust the instructions accordingly to fit your installation.

Setting up the Command Line Environment for CSI_ManageCertificateStore

Typically, CSI_ManageCertificateStore is run as root, but it can also be run by any login that is a member of the cfgsoft group.

To use CSI_ManageCertificateStore the following environment variables must be set:

```
LD_LIBRARY_PATH=/opt/CMAgent/CFC/3.0/lib:/opt/CMAgent/ThirdParty/1.0/lib:$ LD_
LIBRARY_PATH

export LD_LIBRARY_PATH

CSI_REGISTRY_PATH=/opt/CMAgent

export CSI_REGISTRY_PATH

PATH=/opt/CMAgent/CFC/3.0/bin:$PATH

export PATH
```

For HPUX platforms SHLIB_PATH is used in place of LD_LIBRARY_PATH.

For AIX platforms LIBPATH is used in place of LD_LIBRARY_PATH.

CSI_ManageCertificateStore Options

```
[root@localhost tmp]# CSI_ManageCertificateStore -?

Usage: /opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore

-[?h]

[-c certificate_store_name] -[adel] [-g fingerprint] [-s subject] [-f filename]

[-c certificate_store_name] -[iu] -f filename

-h Display this help and exit

-? Display this help and exit

-c The name of the certificate store. This name includes the path.

Defaults to registry value

-a Perform action on all certificates in the store

-d Delete from the certificate store

-e Export certificate(s) and associated key(s) from the certificate store to file(s)
named fingerprint-cert.pem and fingerprint-key.pem ('fingerprint' is the hex SHA1
hash of the certificate)

-f File that contains a certificate external to the certificate store to use. The
certificate in the file must be in PEM format

-g SHA1 hash fingerprint of the certificate in the store to act upon

-i Insert certificate into the certificate store

-k File that contains the private key associated with the certificate. Private
certificate keys are only used for mutual authentication. The key must be in PEM
format. Associating a key with a certificate will cause the registry to be modified
to setup mutual authentication

-l List entries from the certificate store

-n Common name of the certificates in the store to act upon

-p Passphrase for the private key. Needed if the private key PEM

file was passphrase protected, or if the exported key should

be protected

-s Subject of the certificates in the store to act upon
```

```
-u Update certificate in the certificate store
```

Common uses:

Insert a new certificate into the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -i -f filename
```

Update an existing certificate in the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -u -f filename
```

Add a key to an existing certificate in the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -u -f filename -k key_filename
```

Delete an existing certificate from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -d -f filename
```

or

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -d -g fingerprint
```

Delete existing certificates from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -d -s subject
```

Delete all existing certificates from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -d -a
```

Display an existing certificate from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -l -f filename
```

or

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -l -g fingerprint
```

Display existing certificates from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -l -s subject
```

Display all existing certificates from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -l
```

Export an existing certificate and associated key from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -e -f filename
```

or

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -e -g fingerprint
```

Export existing certificates and associated keys from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -e -s subject
```

Export all existing certificates and associated keys from the certificate store:

```
/opt/CMAgent/CFC/3.0/bin/CSI_ManageCertificateStore -e -a
```

CSI_ManageCertificateStore Output

To provide useful feedback to the user CSI_ManageCertificateStore displays information about each certificate that the command acts upon. The displayed information is as follows:

[Action] Certificate:

Fingerprint: *SHA1 hash fingerprint of the certificate*

Common Name: *Common name of the certificate*

Subject : *Subject of the certificate*

CSI_ManageCertificateStore Examples

Following are just a few examples of CSI_ManageCertificateStore use with some additional explanation to give a feel for the tool.

Example of listing certificate store contents

By default the “-l” option for listing certificates will cause all certificates in the store to be listed. This behavior can be modified by specifying options (for example, “-g fingerprint” will always limit the action to the single matching certificate) that narrow the requested results.

```
[root@localhost tmp]# CSI_ManageCertificateStore -l

Certificate:

Fingerprint: 1C564431B9B28DC4D24BB920FD98B539FF57C0C2

Common Name: testca1.VMware.com

Subject : CN = testca1.VMware.com, ST = Colorado, C = US, emailAddress =
ca1@VMware.com, O = VMware, Inc., OU = Testing

Certificate:

Fingerprint: 779403A8D53B1258F3EB09E62A8D17B14CD81DC3

Common Name: Enterprise Certificate 9ACD1B00-42CF-4794-B4E8-B6BDBEC1D4B6
```

```
Subject : O = CSI-SE, OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
9ACD1B00-42CF-4794-B4E8-B6BDBEC1D4B6

Certificate:

Fingerprint: 0041AB5ECF869E1D6A38389A6B834D5768932397

Common Name: Enterprise Certificate 2CA82018-20E1-4487-8A02-DA7A2CFD4304

Subject : O = VMware, Inc., OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
2CA82018-20E1-4487-8A02-DA7A2CFD4304

Certificate:

Fingerprint: 765831AFF8E15332F78D7CBC805F1C68089C8640

Common Name: Enterprise Certificate 7780CB3B-281F-47DF-B48B-5BDE5806C156

Subject : O = QAT, OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
7780CB3B-281F-47DF-B48B-5BDE5806C156
```

Example of deleting a certificate from the store

```
[root@localhost tmp]# CSI_ManageCertificateStore -d -f Enterprise_Certificate_
2CA82018-20E1-4487-8A02-DA7A2CFD4304.pem

Deleting Certificate:

Fingerprint: 0041AB5ECF869E1D6A38389A6B834D5768932397

Common Name: Enterprise Certificate 2CA82018-20E1-4487-8A02-DA7A2CFD4304

Subject : O = VMware, Inc., OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
2CA82018-20E1-4487-8A02-DA7A2CFD4304
```

Note “CSI_ManageCertificateStore -d –g 0041AB5ECF869E1D6A38389A6B834D5768932397” would have produced the same results

Example of inserting a certificate into the store

```
[root@localhost tmp]# CSI_ManageCertificateStore -i -f Enterprise_Certificate_
2CA82018-20E1-4487-8A02-DA7A2CFD4304.pem

Inserting Certificate:

Fingerprint: 0041AB5ECF869E1D6A38389A6B834D5768932397

Common Name: Enterprise Certificate 2CA82018-20E1-4487-8A02-DA7A2CFD4304
```

```
Subject : O =VMware, Inc., OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
2CA82018-20E1-4487-8A02-DA7A2CFD4304
```

Example of exporting certificates from the store

By default the “-e” option for exporting certificates will cause all certificates in the store to be exported. This behavior can be modified by specifying options (for example, “-g fingerprint” will always limit the action to the single matching certificate) that narrow the requested results.

```
[root@localhost example]# CSI_ManageCertificateStore -e

Exporting Certificate:

Fingerprint: 1C564431B9B28DC4D24BB920FD98B539FF57C0C2

Common Name: testca1.VMware.com

Subject : CN = testca1.VMware.com, ST = Colorado, C = US, emailAddress =
ca1@VMware.com, O =VMware, Inc., OU = Testing

Exporting Certificate:

Fingerprint: 779403A8D53B1258F3EB09E62A8D17B14CD81DC3

Common Name: Enterprise Certificate 9ACD1B00-42CF-4794-B4E8-B6BDBEC1D4B6

Subject : O = CSI-SE, OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
9ACD1B00-42CF-4794-B4E8-B6BDBEC1D4B6

Exporting Certificate:

Fingerprint: 0041AB5ECF869E1D6A38389A6B834D5768932397

Common Name: Enterprise Certificate 2CA82018-20E1-4487-8A02-DA7A2CFD4304

Subject : O =VMware, Inc., OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
2CA82018-20E1-4487-8A02-DA7A2CFD4304

Exporting Certificate:

Fingerprint: 765831AFF8E15332F78D7CBC805F1C68089C8640

Common Name: Enterprise Certificate 7780CB3B-281F-47DF-B48B-5BDE5806C156
```

```
Subject : O = QAT, OU = VMware vCenter Configuration Manager, title = VCM
Certificate 7529006C-222F-4EBF-A7E7-F6AB15DB626F, CN = Enterprise Certificate
7780CB3B-281F-47DF-B48B-5BDE5806C156
```

This command produced the following files:

0041AB5ECF869E1D6A38389A6B834D5768932397-cert.pem

1C564431B9B28DC4D24BB920FD98B539FF57C0C2-cert.pem

765831AFF8E15332F78D7CBC805F1C68089C8640-cert.pem

779403A8D53B1258F3EB09E62A8D17B14CD81DC3-cert.pem

If the certificate in the store has an associated private key (this is only used if mutual authentication is set up), an additional file named fingerprint-key.pem will be created. The fingerprint used in the name is the fingerprint of the associated certificate.

VMware, Inc. **3401 Hillview Avenue Palo Alto CA 94304 USA Tel 877-486-9273 Fax 650-427-5001 www.vmware.com**